Introduction au calcul scientifique avec Matlab

TP1

1. Introduction

Pour effectuer ce TP, si vous n'avez pas accès à une licence Matlab sur votre machine, vous pouvez utiliser le logiciel GNU Octave qui est un clone gratuit et libre de Matlab. En règle générale les codes se portent facilement de Matlab vers Octave et vice-versa.

L'évaluation du TP tiendra compte de la rédaction du rapport, qualité des solutions mathématiques, pertinence des commandes utilisées (pensez chemin le plus court pour arriver aux résultats).

De plus, le journal de bord de votre session doit être joint au rapport de la séance.

2. L'aide dans Matlab

Mieux vaut apprendre apprendre à se repérer tout seul que de demander en permanence à son voisin comment faire. Pour cela pensez à l'aide de Matlab

`helpwin`	ouvre une fenêtre contenant la liste des commandes Matlab ainsi que leurs documentations
`help`	donne la liste de toutes les commandes par thèmes
`help` *nom*	décrit la fonction *nom.m*
`lookfor` *nom*	recherche une instruction à partir du mot clé *nom*

3. Commandes générales

3.1. Constantes prédéfinies

`pi`	3.1415...
`eps`	2.2204e-016
`Inf`	nombre infini
`NaN`	n'est pas un nombre ; exprime parfois une indétermination

3.2. Historique

Matlab conserve l'historique des commandes. Il est donc possible de récupérer des instructions déjà saisies (et ensuite de les modifier dans le but de les réutiliser) :

↑, ↓, →, ←	permet de se déplacer dans les lignes de commandes tapées dans la fenêtre de commandes

3.3. Variables d'environnement

Matlab garde en mémoire les variables qui ont été créées. On les voit en haut, à gauche, lorsque Matlab dispose d'une interface graphique. Sinon, on peut les afficher et les effacer par la ligne de commande:

`who`	donne la liste des variables présentes dans l'espace de travail
`whos`	donne la liste des variables présentes dans l'espace de travail ainsi que leurs propriétés
`what`	donne la liste des fichiers `.m` et `.mat` présents dans le répertoire courant
`clear` $var_1 \dots var_n$	efface les variables $var_1, \dots var_n$ de l'espace de travail
`clear`	efface toutes les variables crées dans l'espace de travail

4. Un récapitulatif de quelques commandes utiles

Vecteurs

`n:m`	nombres de `n` à `m` par pas de 1
`n:p:m`	nombres de `n` à `m` par pas de `p`
`linspace(n,m,p)`	`p` nombres de `n` à `m`
`lenght(x)`	longueur de `x`
`x(i)`	i-ème coordonnée de `x`
`x(i1:i2)`	coordonnées `i1` à `i2` de `x`
`x(i1:i2)=[]`	supprimer les coordonnées `i1` à `i2` de `x`
`[x,y]`	concaténer les vecteurs `x` et `y`
`x*y'`	produit scalaire des vecteurs lignes `x` et `y`
`x'*y`	produit scalaire des vecteurs colonnes `x` et `y`
`reshape(x,u,v)`	crée une matrice de taille `[u,v]`, à partir de x

Matrices

`size(A)`	nombre de lignes et de colonnes de `A`
`A(i,j)`	coefficient d'ordre `i,j` de `A`
`A(i1:i2,:)`	lignes `i1` à `i2` de `A`
`A(i1:i2,:) =[]`	supprimer les lignes `i1` à `i2` de `A`
`A(:,j1:j2)`	colonnes `j1` à `j2` de `A`
`A(:,j1:j2)=[]`	supprimer les colonnes `j1` à `j2` de `A`
`A(:)`	indexation linéaire de A, (concaténation des vecteurs colonnes de `A`)
`A(i)`	coefficient d'ordre i dans l'indexation linéaire
`diag(A)`	coefficients diagonaux de `A`

Matrices particulières

`zeros(m,n)`	matrice nulle de taille `m,n`
`ones(m,n)`	matrice de taille `m,n` dont tous les coefficients valent 1
`eye(n)`	matrice identité de taille `n`
`diag(x)`	matrice diagonale dont la diagonale est le vecteur `x`
`magic(n)`	carré magique de taille `n`
`rand(m,n)`	matrice de taille `m,n` à coefficients i.i.d.de loi uniforme sur $[0,1]$
`randn(m,n)`	matrice de taille `m,n` à coefficients i.i.d. de loi normale $\mathcal{N}(0,1)$

Opérations matricielles

`A'`	transposée de `A`
`rank(A)`	rang de `A`
`inv(A)`	inverse de `A`
`expm(A)`	exponentielle de `A`
`det(A)`	déterminant de `A`
`trace(A)`	trace de `A`
`poly(A)`	polynôme caractéristique de `A`
`eig(A)`	valeurs propres de `A`
`[U,D]=eig(A)`	vecteurs propres et valeurs propres de `A`
`+ -`	addition, soustraction
`* ^`	multiplication, puissance (matricielles)
`.* .^`	multiplication, puissance terme à terme
`A\b`	solution de $Ax = b$
`b/A`	solution de $xA = b$
`./`	division terme à terme

5. Exercices

Exercice 1.

Trouvez la fonction qui donne les valeurs propres d'une matrice. Tapez help de cette fonction.

Exercice 2.

Construire :

1. Une suite partant de -8 et allant à -5 par pas de 0.25.
2. Une suite décroissante d'entiers de 15 à 3.
3. Une suite de longueur 100 de –π à π.

Exercice 3.

Ecrire la matrice carrée M d'ordre 12 contenant les entiers de 1 à 144 rangés par ligne. Extraire de cette matrice les matrices suivantes :

1. La sous-matrice formée par les coefficients aij pour i = 1, … , 6 et j = 7, …,12 ;
2. celles des coefficients aij pour (i, j) ϵ 2 {1, 2, 5, 6, 9, 10}^{2} ;
3. celle des coefficients aij pour i + j pair.

Exercice 4.

Trouver une expression Matlab pour construire la matrice suivante :

$$B = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 9 & 7 & 5 & 3 & 1 & -1 & -3 \\ 4 & 8 & 16 & 32 & 64 & 128 & 256 \end{pmatrix}$$

Exercice 5.

Donner une expression Matlab en utilisant une simple multiplication matricielle avec la matrice B pour obtenir :

1. La somme des colonnes 5 et 7 de B
2. La dernière ligne de B
3. Une version de B en interchangeant les lignes 2 et 3

Exercice 6.

Donner une expression Matlab qui en multipliant deux vecteurs permet d'obtenir :

1. La matrice

$$\begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 2 & 3 & 4 & 5 \end{pmatrix}$$

2. La matrice

$$\begin{pmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 2 & 2 & 2 \\ 3 & 3 & 3 \\ 4 & 4 & 4 \end{pmatrix}$$

Exercice 7.

Résolution d'un système sous-dimensionné :

$$\begin{cases} 2x_1 + x_2 - 3x_3 &= 1 \\ x_1 - 2x_2 + x_3 &= 2 \end{cases}$$

1. Ecrivez le système sous la forme matricielle Ax = b (donnez A et b) et calculez le rang de la matrice A.
2. Définissez la matrice B comme étant la matrice A à laquelle on a ajouté le vecteur colonne b.
3. Calculez le rang de la matrice B. Conclusion ?
4. Définissez le vecteur c = [1; 1; 1] et déterminez l'image du vecteur c par la matrice A.
5. Résolvez l'équation Ax = b.

   NB : A\b est équivalent à inv(A)*b si A est inversible.